

# Comments and Q&A for the 2016 Virtual Training: Git, Bitbucket, and GitHub

We are using this page to collect questions and comments on the training.

We won't be able to address everything during the 1.5 hour training, but we will do our best to post responses here!

Another resource to check out is the Software Release Policy Live Q&A, you can browse and submit questions there: <http://feedback.sciencebase.gov/forums/318018-software-release-policy> (Update 11/3/2016: Please use <https://github.com/usgs/best-practices> to browse and submit questions.)

1. **Will this be recorded and available to the public?** Yes, that is the plan. You can view the recording and slides here if logged in to the wiki space: [2016 Virtual Training: Git, Bitbucket, and GitHub](#). Email [cdi@usgs.gov](mailto:cdi@usgs.gov) if you don't have access.
2. **I generally understand GitHub, but I've never used it. I want to feel less intimidated by it.** You are a perfect candidate for this training! We will only briefly touch on GitHub, but a great resource for getting started is the [hello world GitHub Guide](#).
3. **I'm using git a lot but am weak on feature branching and hoping to get more familiar with juggling multiple branches more easily.** Here are some resources that might help:
  - a. <https://git-scm.com/book/en/v2/Git-Branching-Branching-Workflows>
  - b. <https://bocoup.com/weblog/git-workflow-walkthrough-feature-branches>
  - c. <https://www.atlassian.com/git/tutorials/comparing-workflows/feature-branch-workflow>
4. **Start with the basics, including generating and managing your SSH keys.** SSH keys allow you to establish a secure connection between your computer and your git application. If you don't use them, you can still enter your credentials. We aren't covering them in the training, but here is some information
  - a. GitHub: <https://help.github.com/articles/generating-an-ssh-key/>
  - b. Bitbucket: <https://confluence.atlassian.com/bitbucketserver/creating-ssh-keys-776639788.html>
5. **Tip sheets on important processes would be useful.** Here are a few nice cheat sheets (some are geared toward command line)
  - a. <https://www.git-tower.com/blog/git-cheat-sheet/>
  - b. <https://services.github.com/kit/downloads/github-git-cheat-sheet.pdf>
  - c. <http://ndpsoftware.com/git-cheatsheet.html>
6. **Is GitLab similar to use?**
  - a. GitLab is internal USGS only.
  - b. BitBucket is public, with a secured option for USGS, DOI, and partner developers who have myUSGS accounts.
  - c. GitHub is public, with some of the repositories access-restricted within Git's user management system. Anyone approved by the USGS leads can collaborate on the repository.
7. **I'm taking this training to increase proficiency with Github.** We won't be using GitHub directly, but the version control concepts that are presented should benefit your understanding of GitHub.
8. **Can you also cover**
  - a. **view/download access options for projects on bitbucket (public, selected parts public, internal-only?)** On Bitbucket you can make projects or repositories private, accessible to specific people with Bitbucket/myUSGS accounts, or public. An admin would set the permissions on the project level, but you can use the "Settings Repository permissions" page for individual repositories that you have made.
  - b. **whether we can have external collaborators** Approved external collaborators that obtain my.usgs accounts can use Bitbucket. External collaborators are allowed on [GitHub.com](#). External collaborators are not allowed to GitLab.
  - c. **IT security issues raised by automatically pulling code/content from [github.com](#) or any other third party site onto a [usgs.gov](#) server**
9. **If possible, an explanation of appropriate use of public repositories on GitHub would be helpful given our restrictions on release of software (does this include any code?).**
10. One of the items on the [butst page](#) (LH: USGS internal link - I have only successfully accessed it off-campus with VPN) instructs USGS staff to sign up for the "[gs\\_github@usgs.gov](#)" email discussion and announcements group. Perhaps include that in the training. Getting 'coders' to all belong to an email group would really help for communicating changes and updates to supported or recommended systems. At present I'm sensing a lack of coordination between people/groups involved with the 'code management' parallel paths we seem to be on.  
**Ultimately CDI would do well to establish a Confluence sub-site for 'git workflow' and related activities, at least until such time as an authoritative site is established.** Okay all - what do you think? We'll make a call for participation and go from there.
11. **Who hosts BitBucket? - CHS through AWS? Limits? Users admin? costs? Difficulties in existing project migration?** (From Sky) Bitbucket is part of the myUSGS suite of tools and is hosted by Core Science Analytics, Synthesis and Libraries. The whole Atlassian set of tools is currently hosted on USGS servers in Denver. We have looked at Atlassian's own cloud offering (hosted on a server farm in the U.S.), but there are no plans to move in that direction yet. myUSGS, that is now in its fourth generation of technologies since 2005, has

always been open for external collaborators. A few years ago when the Geospatial Information Office (at the time) made ScienceBase an open source project, we applied for and obtained an open source license to run the Atlassian tools with unlimited users. Continuing to push open source projects via myUSGS Bitbucket helps to bolster that continued provision of these tools from Atlassian. There are practical limits in terms of disc space and other resources but no current enforced limits. If the tools are being used to support USGS science, CSASL will continue finding a way to fund them.

One of the beautiful things about Git is that it's pretty simple to move between Git repository spaces. Both GitHub and GitLab offer some additional functionality beyond mere code repo functions (issues, blames, etc.) that are kind of nice. Atlassian's tools have the same ideas, but they are somewhat scattered across the tools (JIRA, Confluence, etc.). If you've taken advantage of things beyond the basics of what Bitbucket does, complete migration to myUSGS might be more challenging.

12. **Are we being encouraged to use BitBucket as our online project VCS repository? (As opposed to projects already being on USGS GitHub and GitLab); What does CDI prefer we use - [GitLab.cr.usgs.gov](https://gitlab.cr.usgs.gov) or Bitbucket?** At the current time, USGS is not suggesting one solution over the others. *CDI funded projects* are required to use Bitbucket in FY16 for a snapshot of code at the end of the project period. Different projects may have different requirements that makes one or another option the best solution, for example, GitHub may be best for social coding and getting discovered by completely external developers, while Bitbucket is able to connect with some other Atlassian and USGS tools.
13. **Re: SourceTree App: What's the significance of the purple "not tracked" icon -- does "not tracked" mean not tracked on the main online repository or ?** In the demo, some files were showing up as "not tracked" because they were new files added to the repository and no action had been taken yet to include it into the "tracked" category.
14. **How does a PI determine what software to release? Example: I have 100 python scripts, how would I know which ones to release?** USGS policy: anything that is in support of your scholarly conclusions must be released. Sky's answer: You should put as much as possible out, it might be useful to others. Sky's addendum: Particularly now with the [U.S. Federal Source Code Policy](#) in effect and the [Open Government Data Act](#) introduced as legislation, open should be the default. However, all of us USGS employees have a responsibility to keep bolstering the USGS reputation for quality science, so follow Fundamental Science Practices and sound software development practices with reviews and tests to make sure your code is a solid foundation for others to use.
15. **Can USGS Bitbucket or GitLab be made accessible to non-USGS people for collaboration with outside agencies?** USGS Bitbucket can be made accessible to collaborators (like all of myUSGS), but GitLab is only accessible on the USGS network (so your collaborators would have to have gone through the process to get credentialed for 2-factor authentication via the VPN).
16. **Where is the Live FAQ [on Software Policy] Sky referred to?** The FAQ is running as an ideation space - <http://feedback.sciencebase.gov/forums/318018-software-release-policy>. Please post your questions. Anyone can chime in with comments. The FSPAC Software Release Policy Subcommittee will review and provide official responses over time.
17. **When moving to a public repository do we really want all of our development commit comments open to the public? Seems like only release versions should be made public. Need more details on software release policy. General discussion don't help those of us actually writing and releasing software.** (From Sky) Absolutely! If you are actively developing software as part of scientific research, the best practice is to use an authenticated code repo and collaborative space limited to your team. There can be serious repercussions to having pre-published scientific deliberations released publicly in any form (commit comments, slides at a conference, etc.). If the information has been released, it precludes the Bureau's ability to withhold the information as pre-decisional in a FOIA situation. That being said, I think there is value to science as a whole in ultimately exposing more of what happens behind the curtain in terms of the experimental and analytical process. Previous versions of software in an analytical workflow and the scientific deliberations along the way are seldom released today from any institution, but they could be important accelerants for scientific discovery as long as the failures are clear. Except for fairly extraordinary situations, the scientific community as a whole does not currently release peer reviews and reconciliation comments, but this is an area of active debate in some parts of the community. I think there is an interesting intersection between the open software development process and peer review in science that can be explored as long as we do that in consultation with supervisors, Science Center Directors, and Bureau Approving Officials. And it does take some degree of extra work to curate the behind the scenes activity for public consumption.